In the Claims:

Please amend the claims as indicated below:

1. (currently amended) A method for identifying an off-schedule software agent operating in a computer system, said method comprising:

associating an entry time with said software agent entering a queue, wherein said queue is a run queue in which said software agent is stored ~~by a manager process~~ in said computer system until an executive process in said computer system is free to process said software agent by running said software agent until said software agent is finished executing, wherein said entry time is a time at which a manager process moves said software agent from a holding queue to said run queue;

obtaining a clock signal associated with a clock time at which said software agent is still stored in said run queue;

comparing said entry time to said clock time to obtain a queue time for said software agent;

comparing said queue time to a threshold limit; and

identifying said software agent as said off-schedule software agent if said queue time exceeds said threshold time limit.

2. (original) The method of claim 1, wherein said clock signal is obtained from a system clock.

3. (original) The method of claim 1, wherein said clock time indicates the current time.

4. (previously presented) The method of claim 1, wherein said threshold time limit is associated with a graded scale for denoting the status of said software agent.

5. (original) The method of claim 1, wherein said threshold time limit is specified by said computer system.

6. (previously presented) The method of claim 1, wherein said software agent is released from said queue if said queue time exceeds said threshold time limit.

7. (previously presented) The method of claim 1, wherein said software agent has a priority associated therewith.

8. (previously presented) The method of claim 7, wherein said priority is changed if said software agent is identified.

9. (previously presented) The method of claim 1, wherein said software agent has information associated therewith, said information allowing statistics of said software agent to be generated.

10. (previously presented) The method of claim 9, wherein said statistics of said software agent are compared to statistics associated with other software agents operating in said queue.

11. (original) The method of claim 9, wherein at least a portion of said information is displayed to a user.

12. (currently amended) A method for managing a plurality of off-schedule software agents concurrently operating in a queue on a computer system, each of said plurality of software agents having data associated therewith, said method comprising:

receiving said data;

processing said data to determine if any of said plurality of off-schedule software agents have excessive queue times, those of said plurality having excessive queue times identified as late software agents, wherein said excessive queue times are determined responsive to a run queue in which said plurality of software agents are stored ~~by a manager process~~ in said computer

system until executive processes in said computer system are free to process respective ones of said plurality of off-schedule software agents by running said off-schedule software agents until said off-schedule software agents are each finished executing, wherein said off-schedule software agents are each determined to be off-schedule responsive to comparing differences between entry times at which a manager process moved each of said off-schedule software agents from a holding queue to said run queue and a later time at which said off-schedule software agents are still stored in said run queue with a threshold time limit associated with said run queue and determining that said differences each exceed said threshold time limit; and

operating on at least said late software agents.

13. (previously presented) The method of claim 12, wherein said operating further comprises:

determining if said late software agents reside in the same database.

14. (previously presented) The method of claim 13, further comprising parsing said late software agents across a plurality of databases.

15. (currently amended) The method of claim 12, wherein said ~~queue has a threshold time limit associated therewith, said~~ threshold time limit is for determining the number of concurrently running software agents allowed to operate in said queue.

16. (previously presented) The method of claim 15, wherein the number of said software agents making up said plurality is compared to said threshold time limit.

17. (original) The method of claim 16, further comprising:

providing a plurality of executive processes if said plurality exceeds said threshold time limit when said comparison is made.

18. (currently amended) The~~A~~ method of claim 12, further ~~for processing data associated with a plurality of off-schedule software agents operating in a computer system, said method~~ comprising:

receiving said data associated with said off-schedule software agents from said run~~a~~ queue ~~associated with said software agents~~ to produce received data, ~~wherein said queue is a run queue in which said software agent is stored by a manager process in said computer system until an executive process in said computer system is free to process said software agent~~;

defining criteria to be used with said received data;

sorting said received data according to said criteria;

generating a list containing said received data;

filtering said received data; and

providing said received data to a document.

19. (original) The method of claim 18, wherein said list is a sorted linked list.

20. (original) The method of claim 19, wherein said filtering removes unwanted agent data.

21. (original) The method of claim 20, wherein said document is made available to a user.

22. (previously presented) The method of claim 21, wherein said document comprises:

instructions for said user to improve operation of at least one of said plurality of software agents.

23. (cancelled)

Please add the following new claims:

24. (new) A computer system including at least one processor and a computer readable memory, said computer readable memory having program code stored thereon for identifying an off-schedule software agent operating in a computer system, said program code comprising:

program code for associating an entry time with said software agent entering a queue, wherein said queue is a run queue in which said software agent is stored in said computer system until an executive process in said computer system is free to process said software agent by

running said software agent until said software agent is finished executing, wherein said entry time is a time at which a manager process moves said software agent from a holding queue to said run queue;

program code for obtaining a clock signal associated with a clock time at which said software agent is still stored in said run queue;

program code for comparing said entry time to said clock time to obtain a queue time for said software agent;

program code for comparing said queue time to a threshold limit; and

program code for identifying said software agent as said off-schedule software agent if said queue time exceeds said threshold time limit.

25. (new) A computer program product, comprising:

a computer readable memory, said computer readable memory having program code stored thereon for identifying an off-schedule software agent operating in a computer system, said program code comprising

program code for associating an entry time with said software agent entering a queue, wherein said queue is a run queue in which said software agent is stored in said computer system until an executive process in said computer system is free to process said software agent by running said software agent until said software agent is finished executing, wherein said entry time is a time at which a manager process moves said software agent from a holding queue to said run queue,

program code for obtaining a clock signal associated with a clock time at which said software agent is still stored in said run queue,

program code for comparing said entry time to said clock time to obtain a queue time for said software agent,

program code for comparing said queue time to a threshold limit, and

program code for identifying said software agent as said off-schedule software agent if said queue time exceeds said threshold time limit.